CLAIMS

Having thus described our invention, what we claim as new and desire to secure by Letters Patent is as follows:

1. A method of executing a linear algebra subroutine, said method comprising:

for an execution code controlling an operation of a floating point unit (FPU) performing a linear algebra subroutine execution, unrolling instructions to prefetch data into a cache providing data into said FPU, said unrolling causing said instructions to touch data anticipated for said linear algebra subroutine execution.

2. The method of claim 1, wherein said prefetching data is accomplished by utilizing time slots caused by a difference between a time to execute instructions in said subroutine execution process and a time to load said data.

3. The method of claim 1, wherein said matrix subroutine comprises a matrix multiplication operation.

4. The method of claim 1, wherein said matrix subroutine comprises a subroutine from a LAPACK (Linear Algebra PACKage).

5. The method of claim 4, wherein said LAPACK subroutine comprises a BLAS Level 3 L1 cache kernel.

6. An apparatus, comprising:

a memory to store matrix data to be used for processing in a linear algebra program;

a floating point unit (FPU) to perform said processing;

a load/store unit (LSU) to load data to be processed by said FPU, said LSU loading said data into a plurality of floating point registers (FRegs); and

a cache to store data from said memory and provide said data to said FRegs,

wherein said matrix data in said memory is touched to be loaded into said cache prior to a need for said data to be in said FRegs for said processing.

7. The apparatus of claim 6, wherein said linear algebra program comprises a matrix multiplication operation.

8. The apparatus of claim 6, wherein said linear algebra program comprises a subroutine from a LAPACK (Linear Algebra PACKage).

9. The apparatus of claim 8, wherein said LAPACK subroutine comprises a BLAS Level 3 L1 cache kernel.

10. The apparatus of claim 6, further comprising:

a compiler to generate instructions for said touching.

11. The apparatus of claim 10, wherein instructions cause a prefetching of said data by utilizing time slots caused by a difference between a time to execute instructions in said subroutine execution process and a time to load said data.

12. A signal-bearing medium tangibly embodying a program of machine-readable instructions executable by a digital processing apparatus to perform a method of executing linear algebra subroutines, said method comprising:

for an execution code controlling an operation of a floating point unit (FPU) performing a linear algebra subroutine execution, unrolling instructions to prefetch data into a cache providing data into said FPU, said unrolling causing said instructions to touch data anticipated for said linear algebra subroutine execution.

13. The signal-bearing medium of claim 12, wherein said prefetching data is accomplished by utilizing time slots caused by a difference between a time to execute instructions in said subroutine execution process and a time to load said data.

14. The signal-bearing medium of claim 12, wherein said matrix subroutine comprises a matrix multiplication operation.

15. The signal-bearing medium of claim 12, wherein said matrix subroutine comprises a subroutine from a LAPACK (Linear Algebra PACKage).

16. The signal-bearing medium of claim 12, wherein said LAPACK subroutine comprises a BLAS Level 3 L1 cache kernel.

17. A method of providing a service involving at least one of solving and applying a scientific/engineering problem, said method comprising at least one of:

using a linear algebra software package that computes one or more matrix subroutines, wherein said linear algebra software package generates an execution code controlling an operation of a floating point unit (FPU) performing a linear algebra subroutine execution, unrolling instructions to prefetch data into a cache providing data into said FPU, said unrolling causing said instructions to touch data anticipated for said linear algebra subroutine execution;

providing a consultation for solving a scientific/engineering problem using said linear algebra software package;

transmitting a result of said linear algebra software package on at least one of a network, a signal-bearing medium containing machine-readable data representing said result, and a printed version representing said result; and

receiving a result of said linear algebra software package on at least one of a network, a signal-bearing medium containing machine-readable data representing said result, and a printed version representing said result.

18. The method of claim 17, wherein said matrix subroutine comprises a subroutine from a LAPACK (Linear Algebra PACKage).

19. The method of claim 18, wherein said LAPACK subroutine comprises a BLAS Level 3 L1 cache kernel.